深度学习认知计算综述

陈伟宏 [1, 2] 安吉尧 [1, 2] 李仁发 [1, 2] 李万里 [1, 2]

摘 要 随着大数据和智能时代的到来, 机器学习的研究重心已开始从感知领域转移到认知计算 (Cognitive computing, CC) 领域, 如何提升对大规模数据的认知能力已成为智能科学与技术的一大研究热点, 最近的深度学习有望开启大数据认知计算领域的研究新热潮. 本文总结了近年来大数据环境下基于深度学习的认知计算研究进展, 分别从深度学习数据表示、认知模型、深度学习并行计算及其应用等方面进行了前沿概况、比较和分析, 对面向大数据的深度学习认知计算的挑战和发展趋势进行了总结、思考与展望.

关键词 深度学习, 认知计算, 张量数据表示, 并行计算, 大数据

引用格式 陈伟宏, 安吉尧, 李仁发, 李万里. 深度学习认知计算综述. 自动化学报, 2017, **43**(11): 1886−1897

DOI 10.16383/j.aas.2017.c160690

Review on Deep-learning-based Cognitive Computing

CHEN Wei-Hong[1, 2] AN Ji-Yao[1, 2] LI Ren-Fa[1, 2] LI Wan-Li[1, 2]

Abstract With the advent of the era of big data and artificial intelligence, the research focus of machine learning has shifted from perception domain to cognitive computing (CC) domain. How to improve the cognitive ability through big data is becoming a research hotspot of intelligence science and technology, in which recent deep learning has been expected to spark a new wave of research on cognitive computing. This paper summarizes the research progress of cognitive computing based on deep learning in recent years. And, comparison and analysis of recent progress in deep learning and cognitive computing are presented from three aspects, that is, deep learning data representation, cognitive models, parallel computing and its applications in the big data environment. Finally, some challenges and development trends of cognitive computing based on deep learning for big data are investigated to for cast the future research.

Key words Deep learning (DL), cognitive computing (CC), tensor data representation, parallel computing, big data

Citation Chen Wei-Hong, An Ji-Yao, Li Ren-Fa, Li Wan-Li. Review on deep-learning-based cognitive computing. *Acta Automatica Sinica*, 2017, **43**(11): 1886−1897

认知计算 (Cognitive computing, CC) 源于模拟人脑的计算机系统的人工智能, 是通过人与自然环境的交互及不断学习, 帮助决策者从不同类型的海量数据中揭示非凡的洞察, 以实现不同程度的感知、记忆、学习和其他认知活动[1]. 随着大数据时代的到来, 丰富的数据和知识为认知计算迎来了新的机遇. 与此同时, 数据的规模、种类、速度和复杂度都远远超过了人脑的认知能力, 如何有效完成对大数据的认知, 给传统认知计算也带来了巨大挑战[2].

认知计算是对新一代智能系统特点的概括. 从功能层面上讲, 认知系统具备人类的某些认知能力, 能够出色完成对数据的发现、理解、推理、决策等特定认知任务[3]. 认知计算是解决理解和学习的问题, 学习能力是认知系统的关键, 特别是在当前大数据时代, 可供学习的数据和知识越来越丰富. 近年来, 得益于计算机硬件性能的提升和云计算技术的发展, 深度学习 (Deep learning, DL) 作为一种新的机器学习方法, 已成为大数据时代认知计算的研究热点之一[4]. 深度学习通过构建基于表示的多层机器学习模型, 训练海量数据, 学习有用特征, 以达到提升识别、分类或预测的准确性[5]. 深度学习可以超越概念学习, 学习到更加复杂的知识, 是深层神经网络学习算法的重大突破[6−7]. 深度学习认知计算是基于深度学习方法挖掘数据中的价值, 以得到更准确、更深层次的知识, 提升对数据的认知能力[8−9]. 基于深度学习的围棋程序 AlphaGo 已达到职业棋手水平[10].

目前, 一个深度学习认知计算过程主要包含三个方面: 深度学习数据表示、深度学习认知模型、深度学习并行计算, 本文综述为三个模块, 如图 1 所示. 认知计算的目标是从输入数据中通过模型学习和

收稿日期 2016-10-10 录用日期 2017-06-22
Manuscript received October 10, 2016; accepted June 22, 2017
国家自然科学基金 (61672217, 61370097) 资助
Supported by National Natural Science Foundation of China (61672217, 61370097)
本文责任编委 张化光
Recommended by Associate Editor ZHANG Hua-Guang
1. 湖南大学信息科学与工程学院 长沙 410082 2. 嵌入式与网络计算湖南省重点实验室 长沙 410082
1. College of Computer Science and Electronic Engineering, Hunan University, Changsha 410082 2. Key Laboratory for Embedded and Network Computing of Hunan Province, Changsha 410082

高性能计算实现分类和理解等认知活动. 大数据环境下, 数据具有大量、多样性、异构性等特征, 如何有效表示数据将直接影响认知模型的建立以及认知计算的效果, 数据表示是深度学习认知计算的基础. 与传统的浅层网络相比, 深层网络模型具有更强大的学习能力, 好的深度学习认知模型将能得到好的认知效果, 模型是关键. 随着大数据集的空前增长, 数据复杂多变, 以及深度模型的复杂, 认知算法处理的是 NP 难问题, 单机计算能力远不能满足深度学习训练的需要, 高性能并行计算成为实现深度学习认知计算的保障.

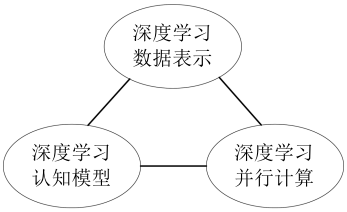


图 1　深度学习认知计算示意图

Fig. 1　Diagram of deep learning cognitive computing

由于深度学习认知算法和计算能力的突破, 深度学习掀起了认知计算领域的一次革命. 目前, 国内外深度学习的研究机构诸如多伦多大学、蒙特利尔大学、斯坦福大学、纽约大学、微软研究院、Google、IBM 研究院、百度公司等, 已将深度学习成功应用于计算机视觉[11]、手写体识别[12]、图像和语音识别[13−15]、音频处理[16−17]、语义表达分类[18]、情感分析[19]、自然语言处理[20−21]、能发现新药物的化学分子分析[22−23] 等多个领域. 深度学习利用深层神经网络学习出高层抽象特征, 这种学习能力与大脑的学习过程相似, 是认知系统的重要组成部分, 已取得了一系列突破性的进展, 但运用深度学习进行认知计算的研究仍处于初级阶段, 深度学习将对认知计算产生深远的影响. 鉴于大数据环境下深度学习认知计算的重要学术研究意义与应用价值, 本文从数据表示、深度学习模型及优化、深度学习并行计算等方面总结现阶段深度学习认知计算的研究进展.

1　深度学习数据表示

多样性作为大数据的重要特征之一, 包含大量异构、有不同分布和复杂关系的大数据已不再是一维或二维的. 相反, 多模态的高维数据越来越常见, 这些数据用传统的矩阵表示不能模型化高度非线性分布, 且容易丢失有用的高阶结构信息. 然而, 深度学习最具有吸引力的特点是它的学习能力. 1962 年, Rosenblatt 给出了著名的学习定理: 人工神经网络可以学会它可以表达的任何东西[24]. 从这个定理可以看出, 人工神经网络的表达能力大大地限制了它的学习能力. 因此, 如何有效表示各种类型数据成为大数据环境下深度学习认知计算研究的重要内容之一[25]. 从数据的不同特征角度看, 数据可分为单模态表示和多模态表示; 从数据的存储形式角度看, 数据可分为向量表示、矩阵表示和张量表示. 本节主要介绍多模态数据表示和张量数据表示方法.

1.1　基于多模态的数据表示

处理大数据多样性的关键是数据集成[26−27]. 针对大数据多样性特点, Ngiam 等[28] 提出多模态数据表示的深度学习方法. 该方法描述了交叉模态特征学习的深度自编码器结构, 使用单模态作为输入, 通过集成音视频数据学习表示, 使用深度自编码器结构来捕获两个模态之间的 “中层” 特征表示. Srivastava 等[29] 提出基于概率的多模态深度玻尔兹曼机 (Deep Boltzmann machine, DBM), 该方法首先在多模态输入的联合空间定义一个概率密度, DBM 为每一个模态数据构建一个栈式受限玻尔兹曼机 (Restricted Boltzmann machines, RBM), 然后用定义的潜在变量状态作为表示, 来学习多模态的有效特征表示. 这种概率形式能将丢失的模态信息从条件概率中采样弥补, 在特征学习阶段, 多模态深度学习比只有一个模态能学到更好的特征.

一个好的多模态学习模型需要满足: 1) 表示空间的相似性隐含对应 “概念” 的相似性; 2) 联合表示在缺乏某些模态情况下也容易获取, 即在给出一定量数据情况下, 填充丢失的模态也是可能的. 多模态深度学习模型一般在通过未标记数据的多模态学习单模态表示和学习共享的数据表示方面效果较好. 大数据环境下, 输入的异构数据包含多个模态, 每个模态有不同的表示和关系结构. 例如, 文本经常表示为离散稀疏的单词计数向量, 使用有真实值和稠密度的像素强度或特征抽取器的输出来表示图像. 这种多模态数据表示难发现存在于不同模态低层特征之间的高度非线性关系, 且比发现相同模态不同特征之间的关系更难, 从而需要构建统一的数据表示方法.

1.2　基于张量的数据表示

多模态深度学习模型在融合共享表示的特征之前, 分别独立地从单个模态学习特征, 它们忽略输入数据空间中异构数据的非线性关系, 导致不能充分学习数据中的有用特征. 基于张量的数据表示方法不同于多模态数据表示, 它以统一方式连接和表示各种异构数据, 能构建统一的数据表示模型化大数据的高阶关系[30−34]. Kuang 等[31] 提出了基于张量的数据表示方法及其处理框架. 数据处理框架包含五大模块: 数据收集模块、数据张量模块、数据降维

模块、数据分析模块和数据服务模块, 如图 2 所示. 数据收集模块负责收集来自不同领域的各种类型的原始数据, 例如音频、视频、XML 文档、GPS 数据等. 这些数据通常为结构化、半结构化或非结构化, 没有统一的数据格式, 基于张量的数据表示为构建统一的数据表示模型提供了新思路. 基于张量的数据表示方法为: 首先根据数据的初始格式生成不同阶数的子张量, 然后融合各子张量成统一的张量表示模型.

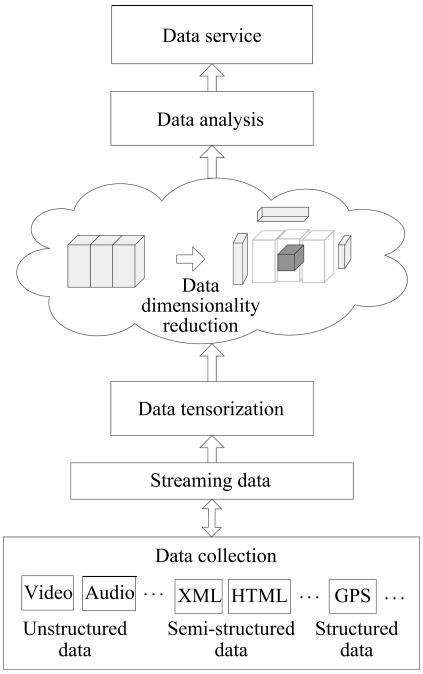


图 2 基于张量的数据表示和处理框架

Fig. 2 Tensor-based data representation and processing framework

1) 基于张量的数据模型

一般地, 时间、空间和数据服务接受者为数据的基本特征, 因此, 定义基于张量的数据模型为

$$T \in \mathbf{R}^{I_t \times I_s \times I_u \times I_1 \times \cdots \times I_p} \tag{1}$$

式中, $(P+3)$ 阶张量包含两部分, 固定部分 $\mathbf{R}^{I_t \times I_s \times I_u}$ 和可扩展部分 $\mathbf{R}^{I_1 \times \cdots \times I_p}$, 张量中 I_t, I_s, I_u 分别表示时间、空间和用户. 在张量模型中, 张量的阶表示数据特征, 异构数据对应张量的阶表示多种特征, 使用张量扩展操作使异构数据的特征依赖于张量模型的固定部分.

2) 张量扩展操作

假设 $A \in \mathbf{R}^{I_t \times I_s \times I_u \times I_1}$, $B \in \mathbf{R}^{I_t \times I_s \times I_u \times I_2}$, 则张量扩展操作为

$$f : A \vec{\times} B \to C,\ C^{I_t \times I_s \times I_u \times I_1 \times I_2} \tag{2}$$

通过式 (2) 可以得出, 首先将异构数据表示成低阶子张量, 然后将其扩展成统一的高阶张量. 张量扩展操作 $\vec{\times}$ 在保留阶的多样性的同时, 融合了相同的张量. 例如, 子张量 T_{sub1} 和 T_{sub2} 有时间阶 I_{t-1}, I_{t-2}, 其中 $I_{t-1} \in \{i_1, i_2\}$, $I_{t-2} \in \{i_1, i_3\}$, 则 $I_{t-1} \vec{\times} I_{t-2} \in \{i_1, i_2, i_3\}$.

3) 统一的张量表示模型

对于非结构化数据 d_u、半结构化数据 d_{semi} 和结构化数据 d_s, 通过下式实现统一的数据张量化操作.

$$f : (d_u \cup d_{\text{semi}} \cup d_s) \to T_u \cup T_{\text{semi}} \cup T_s \tag{3}$$

张量阶可表示成元组形式, 还可描述成可扩展部分和固定部分. 在简化张量成两层的模型中, 内部模型嵌入到三阶 $I_t \times I_s \times I_u$ 整体模型中, 通过张量化方法, 模型化异构数据成子张量, 然后插入到两层模型中得到统一的张量.

在大数据应用中, 数据具有冗余性、不确定性等特点, 要存储所有维度的数据实际上是不可能的, 而获取数据中的价值才是最重要的. 当数据变得非常复杂时, 需要更有效的张量计算方法, 例如用可靠的张量阶估计来处理分布式计算和存储超大规模张量. 对张量数据表示进行分解或维度减少是提升对大数据认知的重要内容[35−36]. Li 等[35] 使用 MapReduce 框架提出了 Tucker 张量分解的可扩展和分布式方法 MR-T, 该方法探索了大数据集的稀疏性来最小化中间数据, 避免了大型矩阵乘, 但 Tucker 分解存在处理高阶张量时面临的维度灾难问题. Zhou 等[36] 基于张量 CP 分解提出了灵活的快速算法 FFCP, 但要求张量数据非常稀疏, 且易于陷入局部收敛. 张量分解算法的高复杂度和收敛速度慢等问题是将其运用于大数据分析的一大瓶颈. 因此, 建立大数据环境下统一高效的张量分解算法是一个研究方向.

2 深度学习认知模型

本节首先介绍深度学习基本思想及典型的深度学习模型, 然后引入基于张量的深度学习模型, 最后比较和分析了深度学习模型优化方法.

2.1 深度学习

深度学习的概念起源于人工神经网络的研究, 早在上世纪 90 年代王飞跃等提出多隐层神经网络模型[37−39], 2006 年 Hinton 等在 *Science* 期刊上提出基于深度信任网 (Deep belief network, DBN) 的非监督训练算法, 实现了深度学习在机器学习研究中的重大突破[8]. 深度学习本质就是简单模块的多层堆叠, 它通过复合多个简单的非线性模块完成学

习过程[6]. 深度学习认知模型如图 3 所示, 其中, L_0 为输入层, L_n 为输出层, w 为连接相邻层之间的权值, $L_1 \sim L_{n-1}$ 为隐层部分, 隐层的输入/输出之间通过非线性函数实现变换.

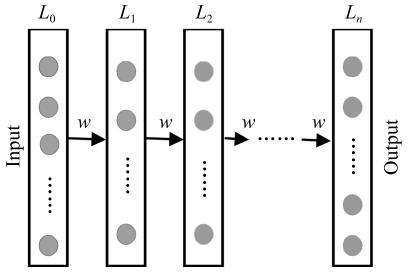


图 3 深度学习一般模型

Fig. 3 The deep learning cognitive model

DNN 是前馈神经网络或具有多隐层的多层感知器, 其计算过程包含前向传播计算和反向传播计算. 从输入层开始, 前向传播计算为: 对第 l 层的每个单元 j, 其值 $y_j^{(l)} = f(z_j^{(l)})$, $z_j^{(l)} = \sum_i w_{ij}{}^{(l)} x_i^{(l)} + b_j^{(l)}$, 其中 i 取值遍历所有第 l 层单元, $0 < l < n$, $x_i^{(l)}$ 为第 l 层输入, $b_j^{(l)}$ 为第 l 层第 j 个单元的偏置项. 一般地, 低一层的输出作为高一层的输入, 即 $x_i^{(l+1)} = y_j^{(l)}$. 反向传播计算采用链式法则, 将误差梯度从高层输出单元传递至低层输入单元, 输出单元的梯度通过对损失函数求导得到. 假设输出单元的损失函数为 $J = 0.5(y^{(l)} - t^{(l)})^2$, 其中 $t^{(l)}$ 为期望输出值, 则

$$\frac{\partial J}{\partial z^{(l)}} = \frac{\partial J}{\partial y^{(l)}} \times \frac{\partial y^{(l)}}{\partial z^{(l)}} = (y^{(l)} - t^{(l)}) f'(z^{(l)})$$

从输出单元到它的前一个隐层, 第 $(l-1)$ 层每个单元 i 的误差梯度计算为

$$\frac{\partial J}{\partial y_i^{(l-1)}} = \sum\nolimits_j \frac{\partial J}{\partial z^{(l-1)}} \times \frac{\partial z^{(l-1)}}{\partial y_j^{(l-1)}} = \sum\nolimits_j w_{ij} \times \frac{\partial J}{\partial z^{(l-1)}}$$

其中, j 取值遍历所有输出层单元. 依此类推, 可得到输入层的误差梯度 $\partial J/\partial x_i$. DNN 一般采用随机梯度下降 (Stochastic gradient descent, SGD) 进行训练. 为了最小化损失函数 J, 每个参数 w_{ij} 和 b_j 都需初始化一个随机值并进行反复迭代, 直到最后收敛. SGD 算法的每一次迭代对参数 w 和 b 的更新计算为

$$w_{ij}^{(l)}(t) = w_{ij}^{(l)}(t-1) - \alpha \frac{\partial J}{\partial w_{ij}^{(l)}}$$

$$b_j^{(l)}(t) = b_j^{(l)}(t-1) - \alpha \frac{\partial J}{\partial b_j^{(l)}}$$

其中, α 为学习率.

深度学习模型中的每个模块变换低层的表示为高层的表示, 逐层复合足够多的变换, 以学到非常复杂的函数. 根据结构不同, 深度学习模型可分为深度神经网络 (Deep neural network, DNN)[6]、深度信任网 DBN[8]、栈式自动编码器 (Stacked auto-encoder, SAE)[40−41]、深度玻尔兹曼机 DBM[42]、卷积神经网络 (Convolutional neural network, CNN)[14]、递归神经网络 (Recurrent neural network, RNN) 等. 对于基于深度学习的认知, 高层表示可以增强某方面的输入, 抑制不相关的变量. 例如, 以数组表示像素值的图像, 在表示的第一层描述图像的特定位置或方向是否有边存在, 第二层检测由边构成的块, 第三层组合这些块构成物体的某一部分, 形成更大的块, 最后组合这些大块成一个整体, 构造出待检测目标[19]. 深度学习的特征层不需要人工设计, 它们通过系统训练自动获取输入的层次特征, 每个模块通过变换它的输入来增加表示的选择性和不变性[43−45].

根据训练数据是否有标记, 深度学习可分为有监督深度学习和无监督深度学习. 有监督的深度学习模型通常训练更有效、构建更灵活, 更适合复杂系统的端到端学习, 例如 DNN、CNN. 深度无监督学习模型更易理解, 更容易嵌入领域知识和处理不确定性问题, 但对复杂系统难于处理推断与学习, 例如 DBN、DBM、RNN、SAE. DBN 可以有效使用未标注数据, 看作一个概率生成模型, 通过产生式预训练方法优化模型的权值, 有效解决认知计算中的过拟合或欠拟合问题, 其时间复杂度与模型大小和深度成线性关系. 使用预训练好的 DBN 来初始化模型权值, 常常会得到比随机初始化方法更好的结果.

2.2 卷积神经网络

卷积神经网络是利用图像的局部特性训练数据, 构造一个部分连通网络, 隐层采用卷积核方式滑动计算的神经网络, 其核心思想是: 局部连接、权值共享、时间或空间采样. 卷积神经网络的层级结构包括数据输入层、卷积层、非线性变换、池化/下采样层和全连接层, 典型的深度卷积网结构 LeNet 如图 4 所示. 卷积层用来检测来自上一层特征的局部连接, 池化层用来融合语义的相似特征. 将卷积、非线性、池化的二个或三个阶段进行堆叠, 然后有更多的卷积和全连接层, 通过网络反向传播梯度, 允许所有卷积核组的权值得到训练.

CNN 训练类似 DNN, 具体过程包含两个阶段: 向前传播和向后传播. 第一阶段, 首先, 从样本中取一个样本 $(X, t^{(l)})$, 将 X 输入网络; 然后经过多次的卷积、非线性变换、池化计算和全连接计算, 得到实际输出 $y^{(l)}$. 第二阶段, 首先计算实际输出 $y^{(l)}$ 与相

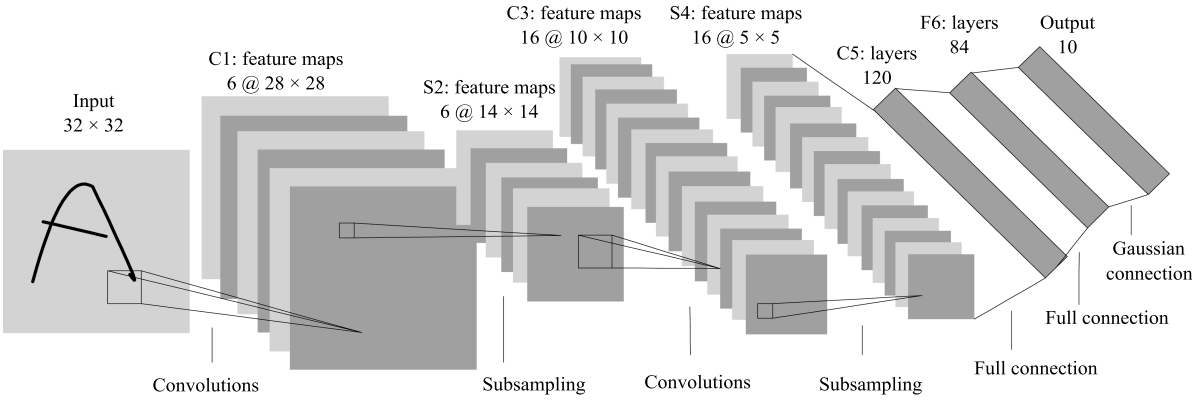


图 4 深度卷积网模型 LeNet

Fig. 4 Deep convolution network model of LeNet

应的期望输出 $t^{(l)}$ 的差; 然后, 按最小化误差方法反向传播调整权值[34]. 卷积层中的单元以特征图形式表示, 每个特征图中使用包含权值集的卷积核 (又称过滤器) 组连接每个单元到前一层特征图的局部块, 然后通过 ReLU 非线性变换传递局部权值和. 假设输入特征图表示为 X, 卷积核为 $K[m,n]$, 则通过卷积计算输出的特征映射 y 为

$$y_{i,j} = (X * K)_{i,j} = \sum_m \sum_n X_{i+m,j+n} K_{m,n} \quad (4)$$

其中, $*$ 为二维离散卷积运算. 所谓池化, 就是图片的下采样. CNN 的池化方法有: 最大池化 (Max pooling)、均值池化 (Mean pooling)、重叠采样 (Overlapping) 等, 其中最经典的是最大池化. 以图 4 为例, 输入大小为 32 像素 × 32 像素的手写体图片, 输出为分类结果, 即 0～9 之间的一个数. C1 为卷积层, 选择 6 个 5 像素 × 5 像素的卷积核, 步长为 1, 可以得到 6 个特征图, 每个特征图大小为 28 × 28, 其中 $28 = 32 - 5 + 1$. S2 为下采样层, 使用最大池化方法, 池大小为 2 像素 × 2 像素, 相当于对 C1 的 28 像素 × 28 像素图片进行分块, 每个块大小为 2 像素 × 2 像素, 取每个块的最大值, 得到 6 张 14 像素 × 14 像素大小的图片. C3 为卷积层, 同样选择卷积核大小为 5 像素 × 5 像素, 步长为 1, 得到新的图片大小为 10 像素 × 10 像素. 这 16 张图片的每一张, 是通过 S2 的 6 张图片与卷积核进行卷积运算后进行加权组合得到. S4 同 S2, 采用 2 像素 × 2 像素大小的池化块进行最大池化计算, 得到 16 张 5 像素 × 5 像素大小的图片. C5 继续用 5 像素 × 5 像素卷积核进行计算, 得到 1 像素 × 1 像素大小的图片, 即一个神经单元. 用 120 个卷积核得到 120 个特征图, 对应 120 个神经单元. 这时神经单元够少了, 后几层用全连接的 DNN 进行处理.

卷积神经网络采用在不同位置共享相同权值, 在二维图像的不同位置检测相同模式, 通过特征图与卷积核的卷积操作大大减少了参数数目, 简化了训练过程. 一个特征图中的所有结果共享相同的卷积核, 但是, 一个卷积核只能提取一个特征, 要提取多个特征就需要多个卷积核, 在同一层中的不同特征图使用不同的卷积核. 相邻池化单元从块中取输入, 这些块通过不超过一行或一列移动减少了描述的维度, 对小的移位或变形创建了不变性. 图像可以直接作为输入, 避免了传统识别算法中复杂的特征抽取和数据重构过程; 权值共享使网络结构更类似于生物神经网络, 减少了权值数量, 降低了网络模型的复杂度. 卷积神经网络可以处理复杂环境下推理规则不明确的问题, 具有良好的容错能力、并行处理能力和自学习能力, 这种网络结构对平移、比例缩放、倾斜等变形具有高度不变性. 目前, 卷积神经网络已大量应用到语音、图像、面部和手势识别等领域.

2.3 栈式自编码器

栈式自编码器是一类典型的无监督深度神经网络, 其输出向量和输入向量同维, 通过隐层学习调整模型参数, 得到每一层权值, 输出不同的表示, 这些表示即为特征. SAE 的主要模块是自编码器, 它由编码器和解码器组成, 如图 5 所示. 自编码器首先经过编码器将输入 x 映射到隐层, 表示为 y; 然后 y 经过解码器又一次映射到隐层, 表示为 z, 通过编码器和解码器可重构输入. 编码器函数形式为

$$y = f(x) = s_f(Wx + \boldsymbol{b}) \quad (5)$$

其中, s_f 为非线性激活函数, 通常为 Sigmoid 函数 $s_f(x) = 1/(1+\mathrm{e}^{-x})$, b 为编码器偏差向量, W 为权值矩阵. 解码器函数形式化为

$$z = g(y) = s_g(W'y + \boldsymbol{b}') \quad (6)$$

其中, s_g 为解码激活函数, 通常为恒等函数或 Sigmoid 函数, b' 为解码器偏差向量, W' 为权值矩阵.

自编码器通过最小化目标函数训练参数 $\{W, b, W', b'\}$, 目标函数为: $J_{AE}(\theta) = \sum_{x\in D} L(x, g(f(x)))$, 其中 L 为二次方误差或交叉熵.

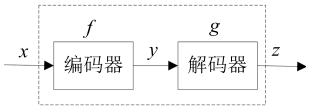


图 5　自编码器结构

Fig. 5　The structure of the auto-encoder

为防止过拟合, 加入权值衰减正则化项到重构误差中: $J_{AE+wd}(\theta) = \sum_{x\in D} L(x, g(f(x))) + \lambda \sum_{ij} w_{ij}^2$, 其中 λ 为超参数, 用于控制正则化强度. 这种加入正则化的 SAE 称为稀疏自编码器 (Sparse auto-encoder, SPAE)[46]. SPAE 采用贪婪算法学习, 时间复杂度和空间复杂度均为线性, 适用于大规模数据的学习. 通过稀疏性约束使一些权值变为 0 的 "Dropout" 方法同时也带来失真, 这种失真将在输入数据或隐层中引入. 例如, Kuang 等[47] 描述的去噪自编码器 (Denoising auto-encoder, DAE), 加入随机噪声到输入数据中. Bengio 等利用二次方重构误差和高斯干扰噪声将结果泛化到了任意参数的编解码过程[48]. 当噪声总量接近于 0 时, 模型可以正确估计生成数据的分布.

2.4　递归神经网络

传统神经网络的输入和输出是相互独立的, 而有些任务的后续输出与前阶段的特征是相关的, 从而引入了具有 "记忆" 概念的递归神经网络[49]. RNN 是深度模型中的隐层节点定向连接成环的人工神经网络. RNN 是一个非常强大的动态系统, 隐层的输出可以存储在存储器中作为另一个输入, 但反向传播的梯度在每个时刻的步骤中既可能增加, 也可能减少. RNN 一旦及时展开, 可看作是非常深度的前馈神经网络, 如图 6 所示. 在隐层单元用节点 s 表示, 在 t 时刻为 s_t, 隐层能取自前一步骤其他神经元的输入. 按照这种方式, 递归神经网络能够映射输入序列 x_t 为输出序列 o_t, 每一个 o_t 依赖于所有前一次的 $x_{t'}$, 其中 $t' \le t$. 在每一步中, RNN 共享了相同的参数 U, V, W.

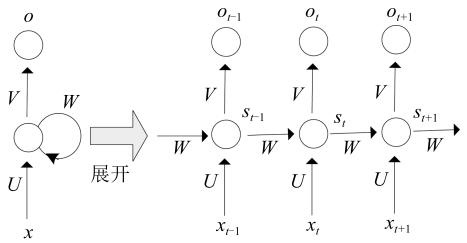


图 6　RNN 结构

Fig. 6　The structure of RNN

Visin 等[50] 提出了 ReNet 结构, 该结构使用四个 RNN 替换 CNN 中的卷积层, 从四个方向扫描低层特征: 从下往上、从上往下、从左到右、从右到左. 与 CNN 相比, RNN 中的递归层确保了它的每一个特征激励为整个图像的特定位置进行激励. 在深层模型训练中, 当前层输出不仅依赖于过去的特征, 还可能依赖未来的特征. 由于标准 RNN 往往忽略了未来的上下文信息, Weninger 等提出了双向 RNN (Bidirectional RNNs, BRNN) 模型[51]. BRNN 的每一步/每个时间点设置多层结构, 每一个训练序列向前和向后是两个 RNN, 连接着同一个输出层, 这种结构为输出层每一节点提供完整的过去和未来上下文信息.

RNN 虽然能够利用深层模型中的上下文信息, 但记忆容量有限, 容易丢失一些连接久远的信息. Wang 等提出了 LSTM (长短期记忆) 的 RNN 方法[52]. LSTM 通过 "门" 让信息选择性通过, 解决了信息长期依赖关系的问题. 目前 LSTM 的几种变体: 增加 "peephole connection", 让门层也接受状态的输入; 使用耦合的忘记门和输入门, 同时做出通过或丢弃的决策[52−53].

2.5　基于张量的深度学习模型

大数据包含大量非结构化、半结构化和结构化数据类型, 多模态数据表示和张量数据表示给出了大数据认知的表示方法, 但如何建立统一的深度学习模型是大数据认知的难点. Yu 等在 DNN 基础上发展了深度张量神经网络 (Deep tensor neural network, DTNN)[54]. DTNN 利用双投影层替换 DNN 中的一个或多个层, 每一个输入向量映射成两个非线性子空间和一个张量层, 两个子空间交互投影来联合预测深度学习的下一层. Deng 等提出张量深度栈式网 (Tensor deep stacked network, T-DSN)[55−56], T-DSN 的每个模块包含两个独立的隐层, 两个独立隐层进行外积运算得到间接隐层, 产生包含两个隐层的所有可能成对的元素乘积的大向量, 这样张量变成了矩阵. 通过构建较大规模的间接隐层 T-DSN, 允许了高阶隐层的特征交互.

Zhang 等基于张量的表示模型, 提出了张量自编码器 (Tensor auto-encoder, TAE) 的张量深度学习模型[57]. TAE 采用张量距离, 而不是欧氏距离和交叉熵作为重构误差中的平均误差平方和, 来激励中间表示以捕获尽可能多的未知高维大数据分布. TAE 设计在张量空间中的高阶 BP (High-order back propagation, HBP) 算法训练模型中参数; 通过堆叠张量自编码器建立张量深度学习模型, 来学习大数据中的多个特征层. 假设 X 为输入张量, f 为 Sigmoid 函数, 分别表示隐层的输入和输出层的

输入, $a^{(2)}_{j_1j_2\cdots j_n}$ $(1 \le j_i \le J_i, 1 \le i \le n)$, $z^{(3)}_{i_1i_2\cdots i_n}$ $(1 \le i_j \le I_j, 1 \le j \le n)$ 分别表示隐层的激励和输出层的激励, 则张量自编码器模型表示为

$$z^{(2)}_{j_1j_2\cdots j_n} = W^{(1)}_{\alpha} \times X + \boldsymbol{b}^{(1)}_{j_1j_2\cdots j_n}, \left(\alpha = j_n + \sum_{i=1}^{N-1}(j_i - 1)\prod_{t=i+1}^{N} J_t\right) \quad (7)$$

$$a^{(2)}_{j_1j_2\cdots j_n} = f\left(z^{(2)}_{j_1j_2\times j_n}\right) \quad (8)$$

$$z^{(3)}_{i_1i_2\cdots i_n} = W^{(2)}_{\beta} \times a^{(2)} + \boldsymbol{b}^{(2)}_{i_1i_2\cdots i_n}, \left(\beta = i_n + \sum_{j=1}^{N-1}(i_j - 1)\prod_{t=j+1}^{N} J_t\right) \quad (9)$$

为了提取大数据特征, 堆叠张量自编码器形成了张量深度学习模型 (Stacking tensor auto-encoder, STAE)[58]. STAE 以栈式自编码器 SAE 相同的方式初始化张量深度学习模型, 使用学习到的第 k 层表示作为第 $k+1$ 层的输入, 仅在前 k 层训练完后训练第 $k+1$ 层. STAE 完成预训练后, 使用它的最高层输出表示作为监督学习算法的输入, 通过调优得到最后的参数. 基于张量的深度学习模型能表示和处理大规模异构数据, 能用于大数据的特征学习, 但是学习大规模异构数据要求更多的参数. 如何对基于张量的深度学习认知模型进行有效的张量分解以减少模型参数或降低算法复杂度, 以及对张量模型认知计算并行化来加速基于深度学习的认知, 值得进一步研究和探讨.

2.6 深度学习模型优化

DNN 中多隐层神经元的使用, 不仅显著提高了 DNN 的建模能力, 而且产生了许多接近最优的配置. 即使参数在学习使用随机梯度下降 (SGD) 方法, 学习过程会陷入局部最优, 但由于出现欠佳的局部最优概率比模型中使用少数神经元时要低, 所以整体效果 DNN 仍然很好. 对于 DNN 学习的高度非凸优化问题, 由于优化是从初始模型开始的, 所以更好的参数初始化技术将会打造出更好的模型. 大数据环境下, 研究者们在构建深度学习模型的同时做了一些模型优化方面的工作, 主要包括非线性变换[59−60]、权值初始化[61−63]、梯度下降[64−65]、性能优化[66−67] 等.

非线性变换. Sigmoid 和 tanh 是 DNN 最常用的非线性单元, 但当网络单元在两个方向都接近饱和时, 梯度变化很小, 整个网络的学习变得很慢. Jaitly 等[68] 为了克服 Sigmoid 单元的缺点, 提出了线性修正单元 (ReLU). 在所有层中随意忽略一部分隐层单元, 这种方法能防止过度拟合, 获得好的效果[44]. 有 ReLU 非线性的神经网络已经证明比标准的 Sigmoid 单元更快, 已被 Dahl 和 Maas 等成功应用在大词汇量语音识别上[69−70]. Salakhutdinov 等提出了在 ReLU 网络中对权值优化的 Path-SGD 方法[61]. 该方法论证了缩放权值对几何性质的不变性, 通过重新考虑合适的几何结构来优化权值, 不影响网络输出. 权值优化方法与几何特性、自适应步长、动量项结合, 在大规模网络上进行更大规模的训练, 将获得更好的效果. Miao 等[71] 提出了 "最大输出" 方法 Maxout, 即在一组固定输入权值上进行最大化操作, 该方法与 CNN 中的最大池化类似. Zhang 等将 Maxout 单元推广为两类: 1) Soft-maxout 将原来的最大化操作替换为 Soft-max 函数; 2) p-norm 单元使用非线性的变换[72]. 实验表明, p-norm 单元使用 $p = 2$ 时, 比 Maxout、tanh 和 ReLU 单元效果都好.

Srivastava 等[73] 提出使用 Dropout 过程, 具体操作如下: 对每个训练实例, 每个隐层单元都随机地以一定概率 (如 $p = 0.5$) 被忽略, 随后除了使用缩放 DNN 权值外, 解码正常完成. Dropout 正则化优点: 训练 DNN 过程使隐层单元仅受自身激励的影响, 而不依赖其他单元, 并提供了一种在不同网络中求其平均模型的方法. 这些优点在训练数据有限或者当 DNN 网络大小比数据要大得多时, 效果更为明显. Dahl 等将 Dropout 策略和 ReLU 单元一起使用, 但仅在全连接的 DNN 高层应用中使用 Dropout[69]. Deng 等[56] 将 Dropout 应用到卷积神经网络的所有层, 包括低层的卷积层和池化层、高层的全连接层, 并发现 CNN 中的 Dropout 率需要大幅降低. 使用 ReLU + Dropout 在 DNN 上训练, 比 Sigmoid 单元相对提高 4.2 %, 比 GMM/ HMM 系统改进 14.4 %[69]. 尽管 Dropout 防止了过度拟合, 有更好的鲁棒性, 但在训练过程中增加了噪声, 降低了学习效率. 更新 Dropout 模式的时刻如何确定, 以及对在大规模数据上训练效果如何, 这些都是关于 Dropout 需进一步研究的问题.

3 深度学习并行计算

并行计算是指把一个原来大而复杂的计算问题, 分解成若干个规模较小且可以应对的子问题, 然后用多个处理器并行地求解这些子问题, 以最终获得原问题的求解[74]. 大数据环境下的深度学习认知计算并行化势在必行. 1) 深度学习的网络模型复杂, 训练数据多, 计算量大. 深度神经网络需要多神经元模拟人的大脑, 神经元之间的连接数量大; 每个神经元包含数学计算, 需要估计的参数量大; DNN 需要大量数据才能训练出高准确率的模型. 2) 深度神经网络需要支持大模型. 例如, 通过增加卷积层的过

滤器数量以及模型深度等, 获得更好的模型质量. 3) 深度神经网络训练难收敛, 需要反复多次实验. 深度神经网络的模型结构、模型初始化等都将影响最终的认知效果. 深度学习认知并行计算, 将极大地加速认知系统对数据的发现、理解、推理和决策. 目前, 已涌现出许多面向深度学习的并行计算方法, 包括 GPU 加速、数据并行与模型并行、计算集群及其并行应用[75−82].

3.1 GPU 加速

超大规模计算支持深度模型, 但同时需要消耗大量的计算资源. 得益于 GPU 众核体系结构, 程序在 GPU 系统上的运行速度相对单核 CPU 往往提升几十倍甚至上千倍. 在大数据训练场景下, 利用 GPU 来训练深度学习模型, 可以充分发挥其高效的并行计算能力, 大幅缩短计算时间. 目前, 深度学习框架的实现集中在基于 GPU 的并行. 在 CUDA GPU 方案中, 每个 GPU 有一个高带宽和高延时的全局存储器, 这种结构允许指令级并行和线程级并行. GPU 并行重在关注数据流, 一方面, 通过转换大数据块减少 RAM 和 GPU 之间的数据传输, 通过上传获得尽可能大的未标记数据, 在全局存储器中存储自由参数; 另一方面, 通过权衡数据并行和学习更新来实现并行[75]. 2015 年 Nvidia 发布了深度学习 GPU 训练系统, 它的 GPU 加速深度学习软件、GPU 加速算法库和 CUDA 深度学习网络, 可以通过更快的复杂模型训练和设计来创造并训练更大、更准确的深度学习模型, 提供加倍优化的深度学习能力.

3.2 数据与模型并行

从训练方式上, 深度学习并行可分为数据并行和模型并行. 数据并行是指对训练数据做切分, 同时采用多个模型实例, 对多个分片的数据并行训练. 要完成数据并行需要做参数交换, 通常由一个参数服务器来协助完成. 模型并行将模型拆分成几个分片, 由几个训练单元分别持有, 共同协作完成训练. 当一个神经元的输入来自另一个训练单元上的神经元的输出时, 产生通信开销. 数据并行和模型并行都不能无限扩展. 数据并行的训练程序太多时, 不得不减小学习率, 以保证训练过程的平稳; 模型并行的分片太多时, 神经元输出值的交换量会急剧增加, 效率大幅下降. 因此, 集成模型并行和数据并行的优势也是一种常见方案.

图 7 是 CNN GPU 的数据并行和模型并行混合架构, 该架构将四个 GPU 分为两组, 组内两个 GPU 做模型并行, 组间做数据并行. 模型并行在计算过程中交换输出值和残差, 数据并行在 Mini-batch 结束后交换模型权重, CNN 数据并行和模型并行框架在图像识别应用中已初见成效, 在微信图像业务中正尝试其众多潜在应用.

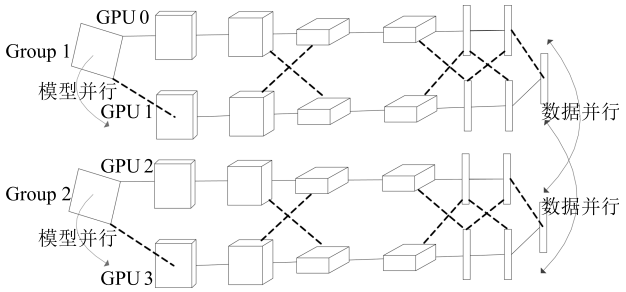


图 7 基于 GPU 的数据并行和模型并行混合架构

Fig. 7 The hybrid architecture based on data parallel and model parallel

3.3 深度学习计算集群及其并行应用

搭建计算集群是加速深度学习的常用方案, 例如 CPU 集群、GPU 集群或异构集群. CPU 集群的优势在于: 利用模型的分布式存储、参数异步通信的特点, 达到大规模分布式计算快速训练的目的. CPU 集群方案适合 GPU 内难以容纳的大模型, 以及稀疏连接的神经网络. Google 搭建的 DistBelief, 是一个采用 CPU 集群的深度学习数据并行和模型并行框架, 为大模型和大数据集的深度学习分布式系统的训练和学习而设计, 实现了随机梯度下降算法的并行化, 支持语音识别和图像分类等认知计算应用[76]. 文献 [76] 将大模型分割成 144 块, 大大提高了训练速度.

结合 GPU 加速技术, 构建 GPU 集群正成为加速大规模深度学习训练的有效解决方案. GPU 集群搭建在 CPU-GPU 异构系统之上, 采用 10 Gbps 等高速的网络通信设施有好的效果. 在开源社区, Krizhevsky 等发布的 Cuda-convnet, 2012 年支持单个 GPU 训练, 2014 年支持多 GPU 上的数据并行和模型并行训练. Jia 等[77] 提供的 Caffe 软件在 CPU 和 GPU 上实现快速卷积神经网络, 使用 NVIDIA K40 或 Titan GPU 在一天内完成 4 千万张图片的训练. 在工业界, Google 的 COTS HPC 系统包括 16 个 GPU 服务器集群, 每个服务器配置四个 NVIDIA GTX680 GPUs, 有 4 GB 存储器, GPU 服务器之间采用 Infiniband 连接, 由 MPI 控制通信[72]. 为了有效使用存储器和 GPU 计算, 实现过程需要仔细设计 CUDA 核. 例如, 计算矩阵乘 $Y = WX$, 其中 W 是过滤器矩阵, X 是输入矩阵, Coates 等[78] 分别利用了矩阵稀疏和局部感受野, 通过对共享同一感受野的神经元抽取 W 中的非零列, 然后与 X 中相应的列相乘, 此策略成功避免要求的存储大于 GPU 共享存储器的情况. Facebook、

百度等结合数据并行和模型并行, 实现了多 GPU 的深度学习训练平台[79−80]. 2014 年, 陈云霁等提出了首个深度学习处理器架构 “寒武纪”, 2017 年提出多芯片机器学习的 “DaDianNiao” 结构来加速深度学习 CNN 和 DNN[81−82]. GPU 集群在发挥单节点高计算能力基础上, 再充分协同集群中多服务器之间的计算能力, 能进一步加速大规模认知计算.

4 深度学习认知计算应用

基于深度学习的认知研究了许多自然信号的分级构成特性. 在分级结构中, 高级特征由低层特征复合构成. 例如, 在图像中, 局部边的复合形成块, 块组成部分, 部分组成物体[5]. 相似的特征存在于语音、词、句、文本等信号中. 2012 年, 斯坦福大学教授 Andrew 和大规模计算顶级专家 Jeff Dean 共同主导的 Google brain 项目运用深度学习技术使得机器系统能学习并自动识别猫[83]. 他们首先利用大量未做标记的图片进行无监督深度学习, 学习获得能够有效表示各种物体的抽象特征; 在此基础上, 利用少量带有标记的训练图像可以快速学习出物体模型. 在 2012 年的 ImageNet 竞赛中, 应用深度卷积网到包含来自 Web 的百万图像的数据集中, 该 Web 包含 1 000 个不同的类, 系统有效使用 GPU、ReLU 和 Dropout 规格化方法, 分类错误率降低至 4.94 %. 2015 年微软亚洲研究院应用深层残差网重构学习过程, 重定向 DNN 中的信息流, 实现了图像的定位、检测与分类, 系统错误率低至 3.57 %[84]. 2015 年, Ren 等基于深度学习技术, 实现了一项机器视觉领域的应用, 即联合卷积神经网络和递归神经网络来生成图像标题[85]. 在语音识别方面, 例如多种语音数据联合训练模型[16], 深度神经网络模型通过逐层训练, 能够提取反映人类语音共性的高层特征, 对目标语音识别产生积极的效果.

与行业解决方案相结合, 基于深度学习的认知计算能从大数据中挖掘出高价值的东西, 实现商业应用. 深度学习认知过程, 通常将其看成黑盒进行训练. 一旦网络训练完成, 可以看透黑盒, 可视化迭代过程中每一层特征图的变化. 如何利用迭代过程中的特征变化规律加速学习, 降低计算成本, 使之面向 CPS 等行业应用是难点.

5 深度学习认知计算发展趋势的思考

得益于 GPU 众核体系结构的出现以及云计算技术的发展, 深度学习以其灵活的深层模型和数据表示优势, 在语音、图像、文本等认知计算领域近几年取得了巨大成功, 这为研究和解决实际认知计算问题带来了新的思想. 然而, 为应对数据的大规模和多样性、全面提升大数据的认知能力, 还存在许多值得深入探讨的问题:

1) 数据表示

深度学习比传统的浅层学习具有更好的非线性表示能力, 但由于海量异构数据的复杂性, 如何构建统一的数据表示来充分学习数据中的有用特征, 是深度学习认知计算中首要解决的问题. 针对超大规模的张量计算问题, 考虑将张量分解、迁移学习注入到深度学习模型中, 对提升问题的认知是一种有效方式.

2) 认知模型

深度学习认知模型在参数数量超过数据点数量时具有完美的样本表达能力. 针对具体行业应用, 如何有效利用学习过程中获得的抽象特征来设计具有更强表达能力的深层模型, 深度学习与强化学习的融合将成为建立轻量级深度学习模型的突破口, 深度学习的黑盒可视化有望开启强人工智能之门.

3) 并行计算

大数据的分布式计算与存储, 以及超大规模基于张量深度学习模型的认知计算问题, 成为大规模异构并行计算的一大难点. 从传统的单 GPU, 到 Google 的异构并行, 再到百度的多核 GPU 并行, 克服了 SGD 不能并行的难题. 然而, 面向有实时性能需求的系统, 如何共享内存、协调异构多处理器之间的计算能力加速大数据环境下的认知, 构建高性能的深度学习认知计算具有十分重要的意义.

4) 深度学习认知计算应用

目前, 深度学习认知计算的应用问题在于是基于深度结构的神经网络灵感模式还是数据特征表示问题. 具备先验知识的智能学习方法在面对新任务时比全新开发的神经网络会表现得更好、训练速度更快. 例如, 基于生成对抗网络 (Generative adversarial networks, GAN) 模型, 用最优传输理论来探索深度神经网络, 看穿机器学习黑盒, 能更好地理解深度学习本质, 从而达到认知计算的要求和目标. 针对特定领域, 例如汽车 CPS、实时嵌入式系统, 如何建立一个轻量计算性能和容易深层理解的深度学习认知模型, 并满足在低计算成本情况下的鲁棒设计需求, 是一个具有挑战性的研究课题.

6 结论

大数据环境下, 基于深度学习的认知计算为近年来大数据分析和理解提供了技术支持. 本文以深度学习认知计算研究进展为依据, 分析了大数据对认知计算带来的问题与挑战, 分三个层面综述了大数据环境下深度学习认知计算的数据表示、认知模型和并行计算, 以及相关应用, 并对基于深度学习的认知计算进行了总结和展望, 希望为相关研究人员提供新思路, 对认知计算产生重要影响.

References

1 Zorzi M, Zanella A, Testolin A, De Filippo, De Grazia M, Zorzi M. Cognition-based networks: a new perspective on network optimization using learning and distributed intelligence. *IEEE Access*, 2015, **3**: 1512−1530

2 Chen Y, Argentinis J, Weber G. IBM Watson: How cognitive computing can be applied to big data challenges in life sciences research. *Clinical Therapeutics*, 2016, **38**(4): 688−701

3 Kelly J. Computing, cognition and the future of knowing [Online], available: https://www.research.ibm.com/software/IBMResearch/multimedia/Computing_Cognition_WhitePaper.pdf, June 2, 2017

4 Yu Kai, Jia Lei, Chen Yu-Qiang, Xu Wei. Deep learning: yesterday, today, tomorrow. *Journal of Computer Research and Development*, 2013, **50**(9): 1799−1804
(余凯, 贾磊, 陈雨强, 徐伟. 深度学习的昨天, 今天和明天. 计算机研究与发展, 2013, **50**(9): 1799−1804)

5 LeCun Y, Bengio Y, Hinton G. Deep learning. *Nature*, 2015, **521**(7553): 436−444

6 Schmidhuber J. Deep learning in neural networks: an overview. *Neural Networks*, 2015, **61**: 85−117

7 Wang F Y, Zhang J J, Zheng X H, Wang X, Yuan Y, Dai X X, Zhang J, Yang L Q . Where does AlphaGo go: from Church-Turing thesis to AlphaGo thesis and beyond. *IEEE/CAA Journal of Automatica Sinica*, 2016, **3**(2): 113−120

8 Hinton G E, Salakhutdinov R R. Reducing the dimensionality of data with neural networks. *Science*, 2006, **313**(5786): 504−507

9 Ludwig L. Toward an integral cognitive theory of memory and technology. *Extended Artificial Memory*. Kaiserslautern: Technical University of Kaiserslautern, 2013. 16−32

10 Silver D, Huang A, Maddison C J, Guez A, Sifre L, van den Driessche G, Schrittwieser J, Antonoglou I, Panneershelvam V, Lanctot M, Dieleman S, Grewe D, Nham J, Kalchbrenner N, Sutskever I, Lillicrap T, Leach M, Kavukcuoglu K, Graepel T, Hassabis D. Mastering the game of Go with deep neural networks and tree search. *Nature*, 2016, **529**(7587): 484−489

11 Ji S W, Xu W, Yang M, Yu K. 3D convolutional neural networks for human action recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2013, **35**(1): 221−231

12 Lake B M, Salakhutdinov R, Tenenbaum J B. Human-level concept learning through probabilistic program induction. *Science*, 2015, **350**(6266): 1332−1338

13 Sun Xu, Li Xiao-Guang, Li Jia-Feng, Zhuo Li. Review on deep learning based image super-resolution restoration algorithms. *Acta Automatica Sinica*, 2017, **43**(5): 697−709
(孙旭, 李晓光, 李嘉锋, 卓力. 基于深度学习的图像超分辨率复原研究进展. 自动化学报, 2017, **43**(5): 697−709)

14 Krizhevsky A, Sutskever I, Hinton G E. ImageNet classification with deep convolutional neural networks. In: Proceedings of the 2012 Advances in Neural Information Processing Systems. Lake Tahoe, Nevada, US: NIPS, 2012. 1097−1105

15 Tompson J, Jain A, LeCun Y, Bregler C. Joint training of a convolutional network and a graphical model for human pose estimation. In: Proceedings of the 2014 Advances in Neural Information Processing Systems. Montreal, Quebec, Canada: NYU, 2014. 1799−1807

16 Yu D, Deng L. *Automatic Speech Recognition: a Deep Learning Approach*. New York: Springer, 2015. 58−80

17 Hinton G E, Deng L, Yu D, Dahl G, Mohamed A, Jaitly N, Senior A, Vanhoucke V, Nguyen P, Kingsbury B, Sainath T. Deep neural networks for acoustic modeling in speech recognition. *IEEE Signal Processing Magazine*, 2012, **29**(6): 82−97

18 Bordes A, Chopra S, Weston J. Question answering with subgraph embeddings [Online], available: http://arxiv.org/abs/1406.3676v3, September 4, 2014

19 Ma J S, Sheridan R P, Liaw A, Dahl G E, Svetnik V. Deep neural nets as a method for quantitative structure-activity relationships. *Journal of Chemical Information and Modeling*, 2015, **55**(2): 263−274

20 He Y H, Xiang S M, Kang C C, Wang J, Pan C H. Cross-modal retrieval via deep and bidirectional representation learning. *IEEE Transactions on Multimedia*, 2016, **18**(7): 1363−1377

21 Jean S, Cho K, Memisevic R, Bengio Y. On using very large target vocabulary for neural machine translation [Online], available: https://arxiv.org/ abs/1412.2007, March 18, 2015

22 Chen Y F, Li Y, Narayan R, Subramanian A, Xie X H. Gene expression inference with deep learning. *Bioinformatics*, 2016, **32**(12): 1832−1839

23 Leung M K K, Xiong H Y, Lee L J, Frey B J. Deep learning of the tissue regulated splicing code. *Bioinformatics*, 2014, **30**(12): i121−i129

24 Xiong H Y, Alipanahi B, Lee L J, Bretschneider H, Merico D, Yuen R K C, Hua Y, Gueroussov S, Najafabadi H S, Hughes T R, Morris Q, Barash Y, Krainer A R, Jojic N, Scherer S W, Blencowe B J, Frey B J. The human splicing code reveals new insights into the genetic determinants of disease. *Science*, 2015, **347**(6218): 1254806

25 Bengio Y, Courville A, Vincent P. Representation learning: a review and new perspectives. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2013, **35**(8): 1798−1828

26 Zheng Y, Capra L, Wolfson O, Yang H. Urban computing: concepts, methodologies, and applications. *ACM Transactions on Intelligent Systems and Technology*, 2014, **5**(3): 1−18

27 Zheng Y. Methodologies for cross-domain data fusion: an overview. *IEEE Transactions on Big Data*, 2015, **1**(1): 16−34

28 Ngiam J, Khosla A, Kim M, Nam J, Lee H, Ng A Y. Multimodal deep learning. In: Proceedings of the 28th International Conference on Machine Learning. Bellevue, Washington, USA: ICML, 2011. 689−696

29 Srivastava N, Salakhutdinov R R. Multimodal learning with deep boltzmann machines. In: Proceedings of the 2012 Advances in Neural Information Processing Systems. Lake Tahoe, Nevada, US: NIPS, 2012. 2222−2230

30 Cichocki A. Era of big data processing: a new approach via tensor networks and tensor decompositions [Online], available: https://arxiv.org/abs/1403.2048, March 9, 2014

31 Kuang L W, Hao F, Yang L T, Lin M, Luo C Q, Min G. A tensor-based approach for big data representation and dimensionality reduction. *IEEE Transactions on Emerging Topics in Computing*, 2014, **2**(3): 280−291

32 Zhang Q C, Yang L T, Chen Z K. Deep computation model for unsupervised feature fearning on big data. *IEEE Transactions on Service Computing*, 2014, **9**(1): 161−171

33 Cohen N, Sharir O, Shashua A. On the expressive power of deep learning: a tensor analysis [Online], available: https://arxiv.org/abs/1509.05009, October 30, 2017

34 Deng L, Yu D. *Deep Learning for Signal and Information Processing*. Delft: Microsoft Research Monograph, 2013. 29−48

35 Li L Z, Boulware D. High-order tensor decomposition for large-scale data analysis. In: Proceedings of the 2015 IEEE International Congress on Big Data. Washington, DC, USA: IEEE, 2015. 665−668

36 Zhou G, Cichocki A, Xie S. Decomposition of big tensors with low multilinearrank [Online], available: https://arxiv.org/pdf/1412.1885.pdf, December 29, 2014

37 Jang J-S R. ANFIS: adaptive-network-based fuzzy inference system. *IEEE Transactions on Systems, Man, Cybernetics*, 1993, **23**(3): 665−685

38 Wang F Y, Kim H. Implementing adaptive fuzzy logic controllers with neural networks: a design paradigm. *Journal of Intelligent and Fuzzy Systems*, 1995, **3**(2): 165−180

39 Wang G, Shi H. TMLNN: triple-valued or multiple-valued logic neural network. *IEEE Transactions on Neural Networks*, 1998, **9**(6): 1099−1117

40 Rosca M, Lakshminarayanan B, Warde-Farley D, Mohamed S. Variational approaches for auto-encoding generative adversarial networks [Online], available: https://arxiv.org/abs/1706.04987, June 15, 2017

41 Lv Y, Duan Y, Kang W, Li Z, Wang F. Traffic flow prediction with big data: a deep learning approach. *IEEE Transactions on Intelligent Transportation Systems*, 2015, **16**(2): 865−873

42 Goodfellow I, Courville A, Bengio Y. Scaling up spike-and-slab models for unsupervised feature learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2013, **35**(8): 1902−1914

43 Chang C H. Deep and shallow architecture of multilayer neural networks. *IEEE Transactions on Neural Networks and Learning Systems*, 2015, **26**(10): 2477−2486

44 Xie K, Wang X, Liu X L, Wen J G, Cao J N. Interference-aware cooperative communication in multi-radio multi-channel wireless networks. *IEEE Transactions on Computers*, 2016, **65**(5): 1528−1542

45 Salakhutdinov R. Learning Deep Generative Models [Ph. D. dissertation], University of Toronto, Toronto, Canada, 2009.

46 Wu X D, Yu K, Ding W, Wang H, Zhu X Q. Online feature selection with streaming features. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2013, **35**(5): 1178−1192

47 Yang L T, Kuang L W, Chen J J, Hao F, Luo C Q. A holistic approach to distributed dimensionality reduction of big data. *IEEE Transactions on Cloud Computing*, 2015, **PP**(99): 1−14

48 Bengio Y, Yao L, Alain G, Vincent P. Generalized denoising auto-encoders as generative models. In: Proceedings of the 26th International Conference on Neural Information Processing Systems. Lake Tahoe, Nevada, USA: ACM, 2013. 899−907

49 Zhu Yu, Zhao Jiang-Kun, Wang Yi-Ning, Zheng Bing-Bing. A review of human action recognition based on deep learning. *Acta Automatica Sinica*, 2016, **42**(6): 848−857
(朱煜, 赵江坤, 王逸宁, 郑兵兵. 基于深度学习的人体行为识别算法综述. 自动化学报, 2016, **42**(6): 848−857)

50 Visin F, Kastner K, Cho K, Matteucci M, Courville A, Bengio Y. ReNet: a recurrent neural network based alternative to convolutional networks [Online], available: https://arxiv.org/abs/1505.00393, July 23, 2015

51 Weninger F, Bergmann J, Schuller B W. Introducing CURRENNT: the munich open-source CUDA recurrent neural network toolkit. *Journal of Machine Learning Research*, 2015, **16**(3): 547−551

52 Wang F, Tax David M J. Survey on the attention based RNN model and its applications in computer vision [Online], available: https://arxiv.org/abs/1601.06823, January 25, 2016

53 Duan Yan-Jie, Lv Yi-Sheng, Zhang Jie, Zhao Xue-Liang, Wang Fei-Yue. Deep learning for control: the state of the art and prospects. *Acta Automatica Sinica*, 2016, **42**(5): 643−654
(段艳杰, 吕宜生, 张杰, 赵学亮, 王飞跃. 深度学习在控制领域的研究现状与展望. 自动化学报, 2016, **42**(5): 643−654)

54 Yu D, Deng L, Seide F. The deep tensor neural network with applications to large vocabulary speech recognition. *IEEE Transactions on Audio, Speech, and Language Processing*, 2013, **21**(2): 388−396

55 Palzer D, Hutchinson B. The tensor deep stacking network toolkit. In: Proceedings of the 2015 International Joint Conference on Neural Networks. Killarney, Ireland: IEEE, 2015. 1−5

56 Deng L, Yu D. Deep learning: methods and applications. *Foundations and Trends in Signal Processing*, 2014, **7**(3−4): 197−387

57 Zhang Q, Yang L T, Chen Z. Deep computation model for unsupervised feature learning on big data. *IEEE Transactions on Services Computing*, 2016, **9**(1): 161−171

58 Hutchinson B, Deng L, Yu D. Tensor deep stacking networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2013, **35**(8): 1944−1957

59 Shan S L, Khalil-Hani M, Bakhteri R. Bounded activation functions for enhanced training stability of deep neural networks on visual pattern recognition problems. *Neurocomputing*, 2016, **216**: 718−734

60 Chandra B, Sharma Rajesh K. Fast learning in deep neural networks. *Neurocomputing*, 2016, **171**: 1205−1215

61 Neyshabur B, Salakhutdinov R R, Srebro N. Path-SGD: path-normalized optimization in deep neural networks. In: Proceedings of the 2015 Advances in Neural Information Processing Systems. Montreal, Canada: Cornell University Library, 2015. 2422−2430

62 Burda Y, Grosse R, Salakhutdinov R. Importance weighted autoencoders [Online], available: https://arxiv.org/abs/1509.00519, November 7, 2016

63 Le Q V, Jaitly N, Hinton G E. A simple way to initialize recurrent networks of rectified linear units [Online], available: https://arxiv.org/abs/1504.00941, April 7, 2015

64 Sutskever I, Martens J, Dahl G, Hinton G. On the importance of momentum and initialization in deep learning. In: Proceedings of the 30th International Conference on International Conference on Machine Learning. Atlanta, GA, USA: ACM, 2013. III-1139−III-1147

65 Kingma D P, Ba J L. Adam: a method for stochastic optimization. In: Proceedings of the 3rd International Conference for Learning Representations. San Diego, CA, USA: Cornell University Library, 2015. 1−15

66 Martens J, Grosse R B. Optimizing neural networks with kronecker-factored approximate curvature. In: Proceedings of the 2015 Neural and Evolutionary Computing. San Diego, CA, USA: Cornell University Library, 2015. 2408−2417

67 Shinozaki T, Watanabe S. Structure discovery of deep neural network based on evolutionary algorithms. In: Proceedings of the 2015 IEEE International Conference on Acoustics, Speech, and Signal Processing. Brisbane, Australia: IEEE, 2015. 4979−4983

68 Jaitly N, Hinton G. Learning a better representation of speech soundwaves using restricted boltzmann machines. In: Proceedings of the 2011 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP). Prague, Czech Republic: IEEE, 2011. 5884−5887

69 Dahl G E, Sainath T N, Hinton G E. Improving deep neural networks for LVCSR using rectified linear units and dropout. In: Proceedings of the 2013 IEEE International Conference on Acoustics, Speech, and Signal Processing. Vancouver, BC, Canada: IEEE, 2013. 8609−8613

70 Maas A L, Hannun A Y, Ng A Y. Rectifier nonlinearities improve neural network acoustic models. In: Proceedings of the 30th International Conference on Machine Learning. Atlanta, USA: IEEE, 2013. 1−6

71 Miao Y J, Metze F, Rawat S. Deep maxout networks for low-resource speech recognition. In: Proceedings of the 2013 IEEE Workshop on Automatic Speech Recognition and Understanding. Olomouc, Czech Republic: IEEE, 2013. 398−403

72 Zhang X H, Trmal J, Povey D, Khudanpur S. Improving deep neural network acoustic models using generalized maxout networks. In: Proceedings of the 2014 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP). Florence, Italy: IEEE, 2014. 215−219

73 Srivastava N, Hinton G E, Krizhevsky A, Sutskever I, Salakhutdinov R. Dropout: a simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 2014, **15**(1): 1929−1958

74 Chen Guo-Liang, Mao Rui, Lu Ke-Zhong. Parallel computing framework of big data. *Chinese Science Bulletin*, 2015, **60**(5−6): 566−569
(陈国良, 毛睿, 陆克中. 大数据并行计算框架. 科学通报, 2015, **60**(5−6): 566−569)

75 Chetlur S, Woolley C, Vandermersch P, Cohen J, Tran J, Catanzaro B, Shelhamer E. cuDNN: efficient primitives for deep learning [Online], available: https://arxiv.org/abs/1410.0759, December 18, 2014

76 Dean J, Corrado G, Monga R, Chen K, Devin M, Mao M, Ranzato M, Senior A, Tuker P, Yang K, Le Q V, Ng A Y. Large scale distributed deep networks. In: Proceedings of the 2012 Neural Information Processing Systems. Lake Tahoe, Nevada, USA: NIPS, 2012. 1223−1231

77 Jia Y Q, Shelhamer E, Donahue J, Darrel T. Caffe: convolutional architecture for fast feature embedding. In: Proceedings of the 22nd ACM International Conference on Multimedia. Orlando, Florida, USA: ACM, 2014. 675−678

78 Coates A, Huval B, Wang T, Wu D, Catanzaro B, Andrew N. Deep learning with COTS HPC systems. In: Proceedings of the 30th International Conference on Machine Learning. Atlanta, Georgia, USA: ICML, 2013. 16−21

79 Yadan O, Adams K, Taigman Y, Ranzato M. Multi-GPU training of convNets [Online], available: https://arxiv.org/abs/1312.5853, February 18, 2013

80 Yu K. Large-scale deep learning at Baidu. In: Proceedings of the 22nd ACM International Conference on Information and Knowledge Management. San Francisco, CA, USA: ACM, 2013. 2211−2212

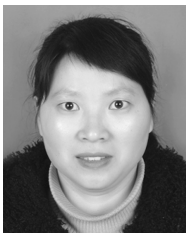
81 Chen Y J, Luo T, Liu S L, Zhang S J, He L Q, Wang J, Li L, Chen T S, Xu Z W, Sun N H, Temam O. DaDianNao: A machine-learning supercomputer. In: Proceedings of the 47th IEEE/ACM International Symposium on Microarchitecture. Cambridge, United Kingdom: IEEE, 2014.

82 Luo T, Liu S L, Li L, Wang Y Q, Zhang S J, Chen T S, Xu Z W, Temam O, Chen Y J. DaDianNao: A neural network supercomputer. *IEEE Transactions on Computers*, 2017, **66**(1): 73−86

83 Markoff J. How many computers to identify a cat? 16,000. *New York Times*, 2012. 6−25

84 He K M, Zhang X Y, Ren S Q, Sun J. Deep residual learning for image recognition. In: Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition. Las Vegas, Nevada, USA: IEEE, 2016. 770−778

85 Ren M Y, Kiros R, Zemel R S. Exploring models and data for image question answering. In: Proceedings of the 28th International Conference on Neural Information Processing Systems. Montreal, Canada: MIT, 2015. 2953−2961



陈伟宏 湖南大学信息科学与工程学院博士研究生, 湖南城市学院教授. 2006 年获得湖南大学硕士学位. 主要研究方向为信息物理系统, 分布式计算, 机器学习.
E-mail: whchen@hnu.edu.cn
(**CHEN Wei-Hong** Ph. D. candidate at the College of Computer Science and Electronic Engineering, Hunan University, and professor at Hunan City University. She received her master degree from Hunan University in 2006. Her research interest covers cyber physical systems, distributed computing, and machine learning.)

安吉尧 湖南大学教授. 2012 年获得湖南大学博士学位. 主要研究方向为信息物理系统, 并行与分布式计算, 计算智能. 本文通信作者.
E-mail: anbobcn@aliyun.com
(**AN Ji-Yao** Professor at Hunan University. He received his Ph. D. degree from Hunan University in 2012. His research interest covers cyber-physical systems, parallel and distributed computing, and computing intelligence. Corresponding author of this paper.)

李仁发 湖南大学教授, 华中科技大学博士. 主要研究方向为嵌入式系统, 信息物理系统, 人工智能与机器视觉.
E-mail: lirenfa@hnu.edu.cn
(**LI Ren-Fa** Professor at Hunan University. He received his Ph. D. degree from Huazhong University of Science and Technology. His research interest covers embedded system, cyber-physical systems, artificial intelligence, and machine vision.)

李万里 湖南大学信息科学与工程学院博士研究生. 2014 年获得湖南大学学士学位. 主要研究方向为机器学习, 计算机视觉, 智能交通系统和驾驶员行为分析.
E-mail: liwanli@hnu.edu.cn
(**LI Wan-Li** Ph. D. candidate at the College of Computer Science and Electronic Engineering, Hunan University. He received his bachelor degree from Hunan University in 2014. His research interest covers machine learning, computer vision, intelligent transportation systems, and driver behavior analysis.)